Bandit Max-Min Fair Allocation

Tsubasa Harada[1] (✉), Shinji Ito[2], and Hanna Sumita[1]

[1] Institute of Science Tokyo, Tokyo, Japan
harada.t.30af@m.isct.ac.jp, sumita@comp.isct.ac.jp
[2] The University of Tokyo, Tokyo, Japan
shinji@mist.i.u-tokyo.ac.jp

Abstract. In this paper, we study a new decision-making problem called the *bandit max-min fair allocation* (BMMFA) problem. The goal of this problem is to maximize the minimum utility among agents with additive valuations by repeatedly assigning indivisible goods to them. One key feature of this problem is that each agent's valuation for each item can only be observed through the semi-bandit feedback, while existing work supposes that the item values are provided at the beginning of each round. Another key feature is that the algorithm's reward function is not additive with respect to rounds, unlike most bandit-setting problems. Our first contribution is to propose an algorithm that has an asymptotic regret bound of $O(m\sqrt{T}\ln T/n + m\sqrt{T\ln(mnT)})$, where n is the number of agents, m is the number of items, and T is the time horizon. This is based on a novel combination of bandit techniques and a resource allocation algorithm studied in the literature on competitive analysis. Our second contribution is to provide the regret lower bound of $\Omega(m\sqrt{T}/n)$. When T is sufficiently larger than n, the gap between the upper and lower bounds is a logarithmic factor of T.

Keywords: Fair allocation · Max-min fairness · Bandit feedback.

1 Introduction

In this paper, we introduce a new sequential decision-making problem, the *bandit max-min fair allocation* (BMMFA) problem, in which some indivisible goods are divided among some agents in a fair manner. The problem is motivated by a problem of designing a subscription service as follows: the company rents items (e.g., clothes, watches, cars, etc.) to users for a certain period, collects the items when the period ends, receives feedback from users, and, based on that feedback, decides which items to rent to whom in the next period. In such a service, the company would like to make all the users as happy as possible. How can we ensure such a fair allocation?

This problem can be regarded as an online variant of the fair allocation problem, which has been a central problem in algorithmic game theory. The classical settings of the fair allocation problem [15] assume that the valuation of each agent for items is known in advance. However, this is not necessarily the case in practice. In the above subscription service, even agents may not recognize their own valuations until they receive items. Therefore, this paper aims to maximize the agents' utilities while learning the valuations of agents through repeatedly allocating items.

We briefly introduce the BMMFA problem. Let $[n] \coloneqq \{1, \dots, n\}$ be a set of n agents with additive valuations, M be a set of m items and T be the time horizon. The value of each agent $i \in [n]$ for each item $e \in M$ follows an unknown distribution D_{ie} over $[0, 1]$ with the expected value μ_{ie}. For each round $t = 1, \dots, T$, the value v_{ie}^t of agent i with respect to an item e is sampled from D_{ie} independently of the round t. We denote by a matrix $a \in \{0, 1\}^{n \times m}$ an allocation of items to agents, where $a_{ie} = 1$ if and only if agent i receives item e. In each round, the algorithm decides an allocation a^t of M based only on the past feedback, and observes values v_{ie}^t only for (i, e) such that $a_{ie}^t = 1$. The utility of agent i obtained at round t (denoted by X_i^t) is the sum of the values for items which are allocated to agent i, i.e., $X_i^t \coloneqq \sum_{e \in M} v_{ie}^t a_{ie}^t$. The utility of agent i at the end of round T is $X_i \coloneqq \sum_{t=1}^T X_i^t$.

As a fairness notion, we adopt the *max-min fairness*, which is a prominent notion in the fair allocation literature [24,3,15]. Then, the sequence of allocation $a^1, \dots, a^T$ is said to be fair if the *egalitarian social welfare*, which is the smallest cumulative utility among agents $\min_{i \in [n]} X_i$, is maximized.

The performance of the algorithm is evaluated by an expected regret R_T, which is the expected difference between the egalitarian social welfare of an optimal policy and that of the algorithm. We assume that an optimal policy chooses a sequence of allocations $x^1, \dots, x^T$ such that $\min_{i \in [n]} \sum_{t=1}^T \sum_{e \in M} \mu_{ie} x_{ie}^t$ is maximized. Therefore, the expected regret R_T is explicitly defined to be

$$R_T \coloneqq \mathbb{E}\left[\min_{i \in [n]} X_i - \min_{i \in [n]} \sum_{t=1}^T \sum_{e \in M} v_{ie}^t x_{ie}^t\right].$$

We have two features in the definition of the regret compared with most other bandit problems: (a) an optimal policy knows all the expected values $\mu_{ie} = \mathbb{E}[v_{ie}^t]$ for all $(i, e) \in [n] \times M$ but may make different allocations across the T rounds, and (b) an algorithm's expected reward is $\mathbb{E}\left[\min_{i \in [n]} X_i\right]$, which is *not additive* with respect to rounds.

To be more specific about (a), a naive definition of an optimal policy would be choosing a fixed allocation $\tilde{x}$ that maximizes $\min_{i \in [n]} \sum_{t=1}^T \sum_{e \in M} \mu_{ie} \tilde{x}_{ie}$. However, this fixed-allocation policy may not be reasonable for our problem. To see this issue, consider the case where $m < n$ and all agents have value 1 for any items. Any fixed-allocation policy has zero egalitarian social welfare since at least one agent receives nothing in every round, while we can achieve positive value by allocating items depending on the round.

For the point (b), $\min_i X_i$ is the fairness measure to be maximized. The problem is that analyzing $\mathbb{E}\left[\min_{i \in [n]} X_i\right]$ is difficult if we naively use the existing bandit techniques. Our model is similar to the combinatorial multi-armed bandit (CMAB) problems [18]. However, even in the most general setting of CMAB, the algorithm's reward is the sum of the per-round rewards ($\min_{i \in [n]} X_i^t$ in our setting), and the optimal policy selects a fixed action for all rounds. This implies that the CMAB framework does not cover our setting.

Another similar allocation problem is studied in the context of competitive analysis [22,31,26]. Roughly speaking, in each round, one item arrives and agents reveals the values for the item, and then the algorithm decides who to receive the item so that

the overall egalitarian social welfare is maximized. However, those resource allocation problems assume that the *full* information $(v_{ie}^t)_{i\in[n],e\in M}$ are given at the *beginning* of each round, whereas only semi-bandit feedback $(v_{ie}^t)_{i,e:a_{ie}^t=1}$ is given at the end of each round in BMMFA. An optimal policy is assumed to know the realization of all the item values in advance, and the performance metrics are defined differently (see Appendix A of the full version [27] for the detail). Therefore, the existing results do not carry to our setting.

1.1 Our Contributions

In this paper, we first define a regret that is suitable for BMMFA. Next, we propose an algorithm that achieves a regret bound of $O(m\sqrt{T}\ln T/n + m\sqrt{T\ln(mnT)})$ when T is sufficiently large. In addition, we provide a lower bound of $\Omega(m\sqrt{T}/n)$ on the regret. The gap between these bounds is $O(\max\{\ln T, n\sqrt{\ln(mnT)}\})$, which is a logarithmic factor of T. Although this paper mainly addresses the case of a known time horizon T, we note that the same regret bound can be achieved even when T is unknown, by using the well-known doubling trick [11]. In the following, we describe the techniques used in the analysis of the regret upper and lower bounds.

Upper Bound Due to the features of our regret definition, it is hard to naively apply the existing approaches. We propose an algorithm by combining techniques of regret analysis and competitive analysis. For this, we employ an idea similar to the resource allocation algorithm proposed in [22] in the context of competitive analysis. This is similar to the multiplicative weight updated method [2]. To estimate the item values given by the semi-bandit feedback, we incorporate upper confidence bounds (UCB) [32] on μ_{ie} for each (i,e), and adopt the error analysis used in [6] for the bandits with knapsacks problem. Our algorithm simply allocates each item to an agent with the largest UCB, discounted by a factor depending on the past allocations. However, the regret analysis is challenging. If we directly analyze the regret, we need to connect the algorithm's choice (depending on UCBs) to the algorithm's reward (in terms of μ_{ie}^t's). This is not easy because the reward is non-additive and UCBs do not imply future item values. We bypass this issue by introducing a *surrogate* regret, defined with expected item values. We show that the original regret and the surrogate differ by at most $O(m\sqrt{T\ln T})$, and the surrogate regret has a bound of $O(m\sqrt{T}\ln T/n + m\sqrt{T\ln(mnT)} + m\ln T\ln(mnT))$. These facts imply an upper bound on R_T. We remark that our algorithm runs in $O(mn)$ time per round.

By this analysis, the average egalitarian social welfare $\mathbb{E}\left[\min_{i\in[n]} X_i/T\right]$ of our algorithm achieves per-round fairness up to an additive error of $o(1)$ when m is fixed. Here, we refer to per-round fairness[3]as maximizing the expected minimum utility per round through a stochastic allocation.

Lower Bound The proof of the lower bound primarily follows the standard method for the multi-armed bandit (MAB) problem by [5]. We first lower bound the regret by averaging over a certain class of instances for BMMFA. Then, by using Pinsker's inequality,

[3] Alternative choices are maximizing the minimum expected ex-ante utility $\min_{i\in[n]}\mathbb{E}[X_i]$ or using only deterministic allocations. In fact, the same guarantee holds for any choice.

we reduce the problem of lower bounding the regret to computing the Kullback-Leibler divergence of certain distributions. The main difference from the standard method for MAB is that we "divide" the problem into m/n subproblems, each with n agents and n items, by treating m/n as an integer. Intuitively, the lower bound $\Omega(m\sqrt{T}/n)$ arises from the number of subproblems times the lower bound of $\Omega(\sqrt{T})$ for each subproblem. This idea of dividing the problem is similar to the proof of the lower bound for the online combinatorial optimization problem [4].

Furthermore, our results are valid for variants of our setting. We will explain this in Section 4.1 of the full version [27].

1.2 Relation to Multi-Player Bandits

The situation of multiple agents choosing items has been actively studied in the context of multi–player bandits (MPB). In this problem, n agents repeatedly choose one of K items (or arms). In the following, we explain the difference between MPB and BMMFA from three perspectives.

The first difference is the correspondence between agents and items: in MPB, each agent chooses exactly one item per round, and there may be items that are not chosen by any agent. In BMMFA, on the other hand, each item is assigned to an agent, and there may be agents who receive no items or multiple items.

The second difference is the objective function: most MPB studies aim to maximize the sum of agents' utilities and do not consider fairness among the agents. See a survey [14] for details. However, some recent studies address the fairness issues [28,30,37,13]. These studies aim to maximize an objective function of the form $\sum_{t=1}^{T} F((X_i^t)_{i\in[n]})$, where $F((X_i^t)_{i\in[n]})$ represents a fairness measure at round t (e.g. *Nash social welfare* [37] or the minimum expected utility over agents [13]). With such objective functions, the algorithm prioritizes per-round fairness rather than overall fairness. In fact, this approach can hinder the achievement of overall fairness because the algorithm lacks an incentive to eliminate the disparity in cumulative utility among agents[4]. On the other hand, in our setting, even if a disparity in utility occurs during the learning process, the algorithm adaptively allocates items to make an agent with small utility happier.

The third perspective involves the differences in the "optimal" policy used as a benchmark for evaluating regret. In the context of bandit problems, including prior studies addressing fairness among agents such as [28,13], the optimal policy typically consists of repeatedly making a single fixed decision. In contrast, in BMMFA, the optimal policy can vary its allocation at each round. In other words, we assume a stronger optimal policy than in similar problems.

These distinctions make it impossible to directly compare the challenges of BMMFA with that of related problems.

[4] Consider an instance with two agents and two goods a and b. The value of a is 1 and that of b is $\varepsilon \ll 1$ for both agents. Any sequence of allocations that gives one item for one agent maximizes $\sum_{t=1}^{T} \min_{i\in[2]} X_i^t$. However, to maximize the minimum of cumulative utilities, we need to assign a to either agent once per two rounds.

1.3 Other Related Work

In the MAB problem, there are K arms, and the algorithm chooses one arm in each round and receives a reward corresponding to the chosen arm. In recent years, there has been research into how to choose an arm that satisfies a certain constraint representing fairness. A commonly used constraint for fairness is that "the ratio of the number of rounds each arm has been drawn to the number of rounds must be greater than a certain value" [34,20,19,36]. In BMMFA, we can view an allocation as an arm. However, as the above notion ignores the utility of agents, it is not suitable for our purpose.

There is a vast body of literature on online fair allocation in combinatorial optimization and algorithmic game theory. Recent studies include problems with a fairness notion such as envy-freeness [10], maximum Nash social welfare [7], p-mean welfare [9,21]. They are just a few examples; see also a survey [1]. Offline sequential allocation problems have also been studied [29,35]. In this context, the goal is to obtain a sequence of allocations with both overall and per-round fairness guarantees.

The one-short, offline version of BMMFA has been studied in combinatorial optimization under the name of the *Santa Clause problem* [24,12,17,23,25]. The problem is NP-hard even to approximate within a factor of better than 1/2 [33]. [8] proposed an $\Omega(\frac{\ln\ln\ln n}{\ln\ln n})$-approximation algorithm for a restricted case. [3] provided the first polynomial-time approximation algorithm for the general problem and this was improved by [25].

Finally, we note that BMMFA can also be viewed as a repeated two-player zero-sum game [16]. Further details can be found in Appendix B of the full version [27].

2 Model

The bandit max-min fair allocation problem is represented by a quadruple $([n], M, T, (D_{ie})_{i\in[n],e\in M})$, where $[n] \coloneqq \{1,\dots,n\}$ is a set of n agents, $M = \{1,\dots,m\}$ is a set of m items, T is the time horizon, and D_{ie} is a probability distribution over $[0,1]$ representing the value of agent i for an item e. For each $i \in [n]$ and $e \in M$, let μ_{ie} be the expected value of D_{ie}. Assume that $[n]$, M and T are known in advance, while $(D_{ie})_{i\in[n],e\in M}$ is not.

Each allocation of items to agents is expressed as an n-row by m-column 0-1 matrix $a \in \{0,1\}^{n\times m}$, where $a_{ie} = 1$ if and only if agent i receives item e in the allocation. Let $\mathcal{A} \subseteq \{0,1\}^{n\times m}$ be a set of allocations, i.e.,

$$\mathcal{A} = \left\{ a \in \{0,1\}^{n\times m} \,:\, \textstyle\sum_{i\in[n]} a_{ie} = 1 \text{ for all } e \in M \right\}.$$

For each round $t = 1,\dots,T$, let v_{ie}^t be a random variable drawn from D_{ie}. Note that the random variables $\{v_{ie}^t : i \in [n], e \in M, t = 1,\dots,T\}$ are mutually independent and are unknown to the algorithm in this step. In round t, the algorithm chooses an allocation $a^t \in \mathcal{A}$ depending only on the previous allocations $(a^s)_{s=1}^{t-1}$ and the feedback obtained by the beginning of round t. Then, the algorithm receives semi-bandit feedback: the algorithm is given the values v_{ie}^t for all (i,e) such that $a_{ie}^t = 1$. The reward of an algorithm ALG is defined by

$$\mathrm{ALG} \coloneqq \min_{i\in[n]} \textstyle\sum_{t=1}^{T} \sum_{e\in M} v_{ie}^t a_{ie}^t.$$

The expected regret R_T is defined to be the expectation of the difference between the egalitarian social welfares of an optimal policy and an algorithm. We assume that an optimal policy takes a sequence of allocations $x^1, \dots, x^T \in \{0,1\}^{n \times m}$ that maximizes the egalitarian social welfare $\min_{i \in [n]} \sum_{t=1}^{T} \sum_{e \in M} \mu_{ie} x_{ie}^t$ with respect to the expected values, i.e., $\min_{i \in [n]} \sum_{t=1}^{T} \sum_{e \in M} \mu_{ie} x_{ie}^t$. Formally, we define

$$\begin{aligned} \mathrm{OPT} &\coloneqq \min_{i \in [n]} \sum_{t=1}^{T} \sum_{e \in M} v_{ie}^t x_{ie}^t, \\ R_T &\coloneqq \mathbb{E}\left[\mathrm{OPT} - \mathrm{ALG}\right]. \end{aligned}$$

For the regret analysis, we introduce surrogate values of OPT and ALG as

$$\begin{aligned} \mathrm{OPT}_\mu &\coloneqq \min_{i \in [n]} \sum_{t=1}^{T} \sum_{e \in M} \mu_{ie} x_{ie}^t, \\ \mathrm{ALG}_\mu &\coloneqq \min_{i \in [n]} \sum_{t=1}^{T} \sum_{e \in M} \mu_{ie} a_{ie}^t \end{aligned}$$

and a *surrogate* regret

$$R_T^\mu \coloneqq \mathbb{E}\left[\mathrm{OPT}_\mu - \mathrm{ALG}_\mu\right].$$

In fact, R_T^μ is not so far from R_T as the following lemma shows. The proof is found in Lemma 1 of the full version [27].

Lemma 1. $|R_T - R_T^\mu| = O(m\sqrt{T \ln T})$.

Furthermore, OPT_μ is upper bounded by the optimal value of the following LP:

$$\begin{aligned} \max_{P,x}\ & T \cdot P \\ \text{s.t. } & P \le \textstyle\sum_{e \in M} \mu_{ie} x_{ie} && (\forall i \in [n]), \\ & \textstyle\sum_{i \in [n]} x_{ie} = 1 && (\forall e \in M), \\ & 0 \le x_{ie} \le 1 && (\forall i \in [n],\ \forall e \in M). \end{aligned} \tag{LP}$$

Indeed, if we set $\hat{x}_{ie} = \sum_{t=1}^{T} x_{ie}^t / T$ ($i \in [n], e \in M$), then $\hat{x}$ is a feasible solution to (LP). Let (P^*, x^*) be an optimal solution of (LP). We will see $T \cdot P^* - \mathbb{E}\left[\mathrm{ALG}_\mu\right]$ to obtain an upper bound on R_T^μ.

Note that (LP) can be interpreted as maximizing the minimum expected per-round utility when a stochastic allocation is allowed. Since P^* upper bounds the maximum "expected minimum" per-round utility, bounding $T \cdot P^* - \mathbb{E}\left[\mathrm{ALG}_\mu\right]$ leads to per-round fairness on average; see Remark 1.

In what follows, we assume $P^* > 0$ because otherwise $R_T^\mu = 0$. Moreover, intuitively, if P^* is sufficiently small, then a per-round utility $\sum_{e \in M} \mu_{ie} a_{ie}^t$ of any agent i is not far less than P^*, and hence ALG_μ is also close to P^*T. Therefore, the difficulty of our problem lies in the case when P^* is large. This is a nature of max-min fair allocation problems. Indeed, existing results in [22,31] for competitive analysis also assume that the offline optimal value is sufficiently large.

3 Algorithm

In this section, we describe an algorithm that has a regret bound of $O(m\sqrt{T} \ln T / n + m\sqrt{T \ln(mnT)} + m \ln T \ln(mnT))$. The regret bound will be shown in the next section. The algorithm is based on the resource allocation algorithm in [22,31]. The brief

description of (a multiple-item variant of) the algorithm is as follows. It is assumed that the values v_{ie}^t for all (i, e) are given at the beginning of each round. Let $\varepsilon > 0$ be a parameter, which will be set later. We denote by u_i^t the cumulative utility of agent i at the end of round t. In each round t, the algorithm chooses an allocation a^t that maximizes a total sum of utilities with respect to item values discounted with u_i. More specifically, a^t achieves $\max_{a \in \mathcal{A}} \sum_{i \in [n], e \in M} (1-\varepsilon)^{u_i^{t-1}/m} v_{ie}^t a_{ie}$.

Due to the feedback model, a direct application of the above resource allocation algorithm is impossible in our setting. It is also not clear whether the existing result carries to due to the different definition of OPT.

To address those issues, we estimate each value using an *upper confidence bound* (UCB), and reconstruct the performance evaluation by incorporating the error analysis used in [6].

For $v \in \mathbb{R}_+$ and $N \in \mathbb{Z}_+$, let $\mathrm{r}(v, N) = \sqrt{C_{\mathrm{rad}} \cdot v/N} + C_{\mathrm{rad}}/N$, where C_{rad} is a positive constant independent of v and N. For each round t and $(i, e) \in [n] \times M$, we define

$$\bar{v}_{ie}^t = \hat{v}_{ie} + \mathrm{r}(\hat{v}_{ie}, N_{ie,t}) \tag{1}$$

as a UCB of v_{ie}^t, where $N_{ie,t}$ is the number of rounds in which item e is assigned to agent i in the first $t-1$ rounds and $\hat{v}_{ie}$ is the average of the $N_{ie,t}$ samples of v_{ie}^t. For this setting of UCBs, the following useful result is known.

Theorem 1 ([6]). *Let $\hat{\nu}$ be the average of N independent samples from a distribution over $[0, 1]$ with expectation ν. For each $C_{\mathrm{rad}} > 0$, it holds that $\Pr[|\nu - \hat{\nu}| \leq \mathrm{r}(\hat{\nu}, N) \leq 3\mathrm{r}(\nu, N)] \geq 1 - e^{-\Omega(C_{\mathrm{rad}})}$. This holds even if $X_1, \ldots, X_N \in [0, 1]$ are random variables, $\hat{\nu} = \frac{1}{N}\sum_{t=1}^N X_t$ is the sample average, and $\nu = \frac{1}{N}\sum_{t=1}^N \mathbb{E}[X_t \mid X_1, \ldots, X_{t-1}]$.*

We will set the constant $C_{\mathrm{rad}} = \Theta(\ln(mnT))$. Then, by using the union bound, we have

$$\mu_{ie} \in [\hat{v}_{ie}^t - \mathrm{r}(\hat{v}_{ie}^t, N_{ie,t}), \hat{v}_{ie}^t + \mathrm{r}(\hat{v}_{ie}^t, N_{ie,t})]$$

for any $(i, e) \in [n] \times M$ and round t with probability at least $1 - \frac{1}{T}$. We call this event a *clean execution* [6] and denote it by $\mathcal{E}$.

Our algorithm is summarized in Algorithm 1. We devote the first n rounds to collect one sample of each item value. At the subsequent rounds t, assuming $\bar{v}_{ie}^t$ as an estimation of μ_{ie}, we choose an allocation a^t maximizing $\sum_{i \in [n], e \in M} (1-\varepsilon)^{u_i^{t-1}/m} \bar{v}_{ie}^t \cdot a_{ie}^t$. We can obtain a^t easily just by allocating each item e to the agent with the largest discounted UCB for e.

4 Regret Analysis

The goal of this section is to prove the following theorems, which provides an asymptotic guarantee on R_T.

Theorem 2. *The regret R_T for Algorithm 1 is bounded as*

$$R_T \leq m + \varepsilon W' + n \cdot e^{-\frac{\varepsilon^2 W'}{2m}} W' + P^* + O(err),$$

Algorithm 1 Allocation algorithm

Parameter: $\varepsilon \in (0, 1)$.

1: **for** $t = 1, \ldots, n$ **do**
2: Assign all items to agent t and receive values v_{te}^t for each $e \in M$.
3: **end for**
4: Set $\bar{v}_{ie}^{n+1}$ as in (1) for each $i \in [n]$ and $e \in M$.
5: Let $u_i^n = 0$ for each $i \in [n]$.
6: **for** $t = n + 1, \ldots, T$ **do**
7: Let a^t be an allocation $a \in \mathcal{A}$ maximizing

$$\sum_{i \in [n]} \sum_{e \in M} (1 - \varepsilon)^{\frac{1}{m} u_i^{t-1}} \bar{v}_{ie}^t \cdot a_{ie}.$$

8: Receive values v_{ie}^t for each (i, e) such that $a_{ie}^t = 1$.
9: Set $u_i^t \leftarrow u_i^{t-1} + \sum_{e \in M} \bar{v}_{ie}^t a_{ie}^t$ for each $i \in [n]$.
10: Set $\bar{v}_{ie}^{t+1}$ accordingly as in (1).
11: **end for**

where $W' = P^*(T-n)$, $err = O(\sqrt{C_{\mathrm{rad}} m^2 T} + C_{\mathrm{rad}} m \ln T)$ *and* $C_{\mathrm{rad}} = \Theta(\ln(mnT))$. *If* $T \geq e^{\frac{2m}{P^*}} + n$, *by setting* $\varepsilon = \ln(T-n)/\sqrt{T-n}$, *we have* $R_T = O(m\sqrt{T} \ln T/n + err)$.

Weakening the assumption on T to $T \geq n$ yields another regret bound.

Theorem 3. *The regret R_T for Algorithm 1 is also bounded as*

$$R_T \leq \frac{m\varepsilon T}{n} + \frac{m \ln n}{\varepsilon} + O(err).$$

If $T \geq n$, *by setting* $\varepsilon = \sqrt{n \ln n / T}$, *we have* $R_T = O(m\sqrt{T \ln n/n} + err)$.

By Lemma 1, we have $R_T \leq R_T^\mu + O(m\sqrt{T \ln T}) = R_T^\mu + O(err)$. Then, to prove Theorems 2 and 3, it suffices to show the upper bound on the surrogate regret R_T^μ.

First, we prove Theorem 2. As described before, $R_T^\mu \leq TP^* - \mathbb{E}[\mathrm{ALG}_\mu]$. For each agent i, let X_i^t be random variables representing the reward of the agent i at round t with respect to the expected item values, i.e., $X_i^t = \sum_{e \in M} \mu_{ie} a_{ie}^t$. For notational convenience, let $W' = P^*(T-n)$ and let $\mathrm{ALG}'_\mu = \min_{i \in [n]} \sum_{t=n+1}^T X_i^t$. The following simple calculations allow us to ignore regret in the first n rounds:

$$\begin{aligned} R_T^\mu &\leq P^*T - \mathbb{E}\left[\min_{i \in [n]} \textstyle\sum_{t=n+1}^T X_i^t\right] \\ &= P^*n + W' - \mathbb{E}\left[\mathrm{ALG}'_\mu\right] \leq m + W' - \mathbb{E}\left[\mathrm{ALG}'_\mu\right]. \end{aligned} \tag{2}$$

Then, in the rest of this section, we bound $W' - \mathbb{E}\left[\mathrm{ALG}'_\mu\right]$.

For each agent i, let $\bar{X}_i^t$ be a random variable representing the reward of the agent i at round t if values are replaced with their UCBs, i.e., $\bar{X}_i^t = \sum_{e \in M} \bar{v}_{ie}^t a_{ie}^t$. In addition, let $\overline{\mathrm{ALG}'} \coloneqq \min_{i \in [n]} \sum_{t=n+1}^T \bar{X}_i^t$ be the total reward of Algorithm 1 after round n

with respect to the UCBs. We first claim that $\mathbb{E}\left[\overline{\mathrm{ALG}'}\right]$ is not far from $\mathbb{E}\left[\mathrm{ALG}'_\mu\right]$ in the follwing lemma. Let $err \coloneqq \mathbb{E}\left[\overline{\mathrm{ALG}'}\right] - \mathbb{E}\left[\mathrm{ALG}'_\mu\right]$.

Lemma 2. $err = \mathbb{E}\left[\overline{\mathrm{ALG}'}\right] - \mathbb{E}\left[\mathrm{ALG}'_\mu\right] = O(\sqrt{C_{\mathrm{rad}} m^2 T} + C_{\mathrm{rad}} m \ln T)$.

The proof of Lemma 2 is similar to the proof of Lemma 5.6 in [6] and is given in Lemma 2 of the full version [27]. Lemma 2 implies that we only need to evaluate $W' - \mathbb{E}\left[\overline{\mathrm{ALG}'}\right]$. We proceed based on the idea in [22,31].

By the union bound and Markov's inequality, the probability that $\overline{\mathrm{ALG}'}$ is at most $(1-\varepsilon)W'$ is

$$\begin{aligned}
&\Pr\left[\min_{i\in[n]} \sum_{t=n+1}^{T} \bar{X}_i^t \le (1-\varepsilon)W'\right] \\
&\le \sum_{i\in[n]} \Pr\left[\sum_{t=n+1}^{T} \bar{X}_i^t \le (1-\varepsilon)W'\right] \\
&= \sum_{i\in[n]} \Pr\left[(1-\varepsilon)^{\frac{1}{m}\sum_{t=n+1}^{T} \bar{X}_i^t} \ge (1-\varepsilon)^{\frac{(1-\varepsilon)W'}{m}}\right] \\
&\le \sum_{i\in[n]} \mathbb{E}\left[(1-\varepsilon)^{\frac{1}{m}\sum_{t=n+1}^{T} \bar{X}_i^t}\right] / (1-\varepsilon)^{\frac{(1-\varepsilon)W'}{m}}. \qquad (3)
\end{aligned}$$

If the rightmost value in (3) is sufficiently small, then we can bound the regret by $m + O(\varepsilon W')$ with high probability. For $s = n, n+1, \ldots, T$, let us define $\Phi(s)$ as

$$\Phi(s) \coloneqq \sum_{i\in[n]} (1-\varepsilon)^{\frac{1}{m}\sum_{t=n+1}^{s} \bar{X}_i^t} \cdot \left(1 - \frac{\varepsilon P^*}{m}\right)^{T-s}.$$

We note that the rightmost value in (3) is equal to $\mathbb{E}[\Phi(T)]/(1-\varepsilon)^{(1-\varepsilon)W'/m}$.

Lemma 3. *In a clean execution of Algorithm 1, $\Phi(s)$ is monotone non-increasing in s.*

Proof. Since the feasible region of (LP) is a subset of the convex hull of some integral allocations of M to $[n]$, we can decompose x^* as a convex combination of integral allocations $y^1, \ldots, y^k$ so that $x^* = \sum_{j\in[k]} \lambda_j y^j$, where $\lambda_j \ge 0$ ($\forall j \in [k]$) and $\sum_{j\in[k]} \lambda_j = 1$. We note that $y^1, \ldots, y^k$ are not necessarily optimal solutions to (LP). Then, for $s = n+1, \ldots, T-1$, letting $\alpha_i = (1-\varepsilon)^{\frac{1}{m}\sum_{t=n+1}^{s} \bar{X}_i^t}$, we can see that

$$\begin{aligned}
\Phi(s+1) &= \sum_{i\in[n]} \alpha_i \cdot (1-\varepsilon)^{\frac{1}{m}\bar{X}_i^{s+1}} \cdot \left(1 - \frac{\varepsilon}{m} P^*\right)^{T-s-1} \\
&\le \sum_{i\in[n]} \alpha_i \cdot \left(1 - \frac{\varepsilon}{m} \bar{X}_i^{s+1}\right) \cdot \left(1 - \frac{\varepsilon}{m} P^*\right)^{T-s-1} \\
&\le \sum_{i\in[n]} \alpha_i \cdot \left(1 - \frac{\varepsilon}{m} P^*\right) \cdot \left(1 - \frac{\varepsilon}{m} P^*\right)^{T-s-1}
\end{aligned}$$

$$= \sum_{i\in[n]} \alpha_i \cdot \big(1 - \frac{\varepsilon}{m} P^*\big)^{T-s} = \Phi(s).$$

Here, the first inequality holds by $\bar{X}_i^{s+1}/m \in [0,1]$ and $(1-\varepsilon)^x \leq 1 - \varepsilon x$ for any $x \in [0,1]$. As for the second inequality,

$$\begin{aligned}\textstyle\sum_{i\in[n]} \alpha_i \cdot \bar{X}_i^{s+1} &\geq \textstyle\sum_{j=1}^{k} \lambda_j \sum_{i\in[n]} \alpha_i \sum_{e\in M} \bar{v}_{ie}^{s+1} y_{ie}^j \\ &= \textstyle\sum_{i\in[n]} \alpha_i \sum_{e\in M} \bar{v}_{ie}^{s+1} x_{ie}^*\end{aligned}$$

holds for each $i \in [n]$ by the choice of a^t in line 7. Since we assume a clean execution, it further holds that $\sum_{e\in M} \bar{v}_{ie}^{s+1} x_{ie}^* \geq \sum_{e\in M} \mu_{ie} x_{ie}^* \geq P^*$.

The proof of Lemma 3 requires a connection between a utility with respect to the UCBs and an optimal policy. This task is made easier if we use the surrogate regret.

Lemma 4. $\Phi(n)/(1-\varepsilon)^{(1-\varepsilon)W'/m} \leq n \cdot e^{-\frac{\varepsilon^2 W'}{2m}}$.

Proof. By Lemma 3 and $1 - x \leq e^{-x}$ for any x, we have

$$\frac{\Phi(n)}{(1-\varepsilon)^{\frac{(1-\varepsilon)W'}{m}}} = \frac{\sum_{i\in[n]} \big(1 - \frac{\varepsilon P^*}{m}\big)^{T-n}}{(1-\varepsilon)^{(1-\varepsilon)\frac{W'}{m}}} \leq \frac{n \cdot e^{-\varepsilon \frac{W'}{m}}}{(1-\varepsilon)^{(1-\varepsilon)\frac{W'}{m}}}.$$

This is bounded by $n \cdot e^{-\frac{\varepsilon^2 W'}{2m}}$ since $\frac{1}{(1-\varepsilon)^{(1-\varepsilon)}} \leq e^{\varepsilon - \varepsilon^2/2}$ for any $\varepsilon \in [0,1)$.

Now we are ready to prove theorems.

Proof (Proof of Theorem 2). By applying Lemmas 3 and 4 to (3), we see that

$$\begin{aligned}\Pr\Big[\overline{\mathrm{ALG}'} \leq (1-\varepsilon)W'\Big] &\leq \Pr\Big[\overline{\mathrm{ALG}'} \leq (1-\varepsilon)W' \mid \mathcal{E}\Big] + \frac{1}{T} \\ &\leq \frac{\mathbb{E}[\Phi(T) \mid \mathcal{E}]}{(1-\varepsilon)^{(1-\varepsilon)W'/m}} + \frac{1}{T} \\ &\leq \frac{\Phi(n)}{(1-\varepsilon)^{(1-\varepsilon)W'/m}} + \frac{1}{T} \leq n \cdot e^{-\frac{\varepsilon^2 W'}{2m}} + \frac{1}{T}.\end{aligned}$$

This implies that

$$\begin{aligned}W' - \mathbb{E}\Big[\overline{\mathrm{ALG}'}\Big] &\leq \varepsilon W' + (n \cdot e^{-\frac{\varepsilon^2 W'}{2m}} + 1/T)W' \\ &\leq \varepsilon W' + n \cdot e^{-\frac{\varepsilon^2 W'}{2m}} W' + P^*. \qquad (4)\end{aligned}$$

This together with (2) and Lemma 2 implies that

$$R_T^{\mu} \leq m + \varepsilon W' + n \cdot e^{-\frac{\varepsilon^2 W'}{2m}} W' + P^* + O(err). \qquad (5)$$

Let $T' := T - n$ ($\geq e^{\frac{2m}{P^*}}$), and we set $\varepsilon = \frac{\ln T'}{\sqrt{T'}}$. Then it follows that $\varepsilon W' + n \cdot e^{-\frac{\varepsilon^2 W'}{2m}} W' = P^* \sqrt{T'} \ln T' + n P^* T'^{1 - \frac{P^*}{2m} \ln T'} \leq P^* \sqrt{T'} \ln T' + n P^*$. Therefore, from (5), we finally see that

$$R_T \leq m + P^* \sqrt{T'} \ln T' + n P^* + P^* + O(err) = O\Big(\frac{m}{n} \sqrt{T} \ln T + err\Big).$$

This completes the proof of Theorem 2.

Proof (Proof of Theorem 3). Let $i^* \in \arg\min_i \sum_{t=n+1}^{T} \bar{X}_i^t$. Under the clean execution $\mathcal{E}$, we have

$$(1-\varepsilon)^{\frac{1}{m}\sum_{t=n+1}^{T} \bar{X}_{i^*}^t} \leq \sum_{i\in[n]} (1-\varepsilon)^{\frac{1}{m}\sum_{t=n+1}^{T} \bar{X}_i^t} = \Phi(T) \leq \Phi(n) = n\left(1-\frac{\varepsilon P^*}{m}\right)^{T-n},$$

where the inequality follows from from Lemma 3. By taking the logarithm of both sides, it follows that $\frac{1}{m}\sum_{t=n+1}^{T} \bar{X}_{i^*}^t \ln(1-\varepsilon) \leq (T-n)\ln\left(1-\frac{\varepsilon P^*}{m}\right) + \ln n$. As we have $-x-x^2 \leq \ln(1-x) \leq -x$ for $x \leq 1/2$, we obtain

$$\left(-\varepsilon-\varepsilon^2\right)\frac{1}{m}\sum_{t=n+1}^{T} \bar{X}_{i^*}^t \leq -(T-n)\frac{\varepsilon P^*}{m} + \ln n = -\frac{\varepsilon W'}{m} + \ln n.$$

Therefore, under the clean execution $\mathcal{E}$, it follows that

$$W' - \overline{\mathrm{ALG}'} = W' - \sum_{t=n+1}^{T} \bar{X}_{i^*}^t \leq \varepsilon \sum_{t=n+1}^{T} \bar{X}_{i^*}^t + \frac{m}{\varepsilon}\ln n \leq \frac{m}{n}\varepsilon T + \frac{m}{\varepsilon}\ln n.$$

Since $\mathcal{E}$ occurs with probability at least $1-1/T$, we have

$$W' - \mathbb{E}\left[\overline{\mathrm{ALG}'}\right] \leq \frac{m}{n}\varepsilon T + \frac{m}{\varepsilon}\ln n + P^*(T-n)/T. \tag{6}$$

Applying (6) to the proof of Theorem 2 instead of (4) yields Theorem 3.

Remark 1. We observe the outcome of Algorithm 1 almost achieves per-round fairness on average across rounds. Here we mean per-round fairness by attaining the maximum expected minimum utility per round with a stochastic allocation, whose value is bounded by P^*. Indeed, by (4), we have $P^* - \mathbb{E}\left[\min_{i\in[n]} \frac{1}{T}\sum_{t=1}^{T} X_i^t\right] \leq P^* - \frac{1}{T}\mathbb{E}\left[\mathrm{ALG}'_\mu\right] = P^*(\varepsilon + n\cdot e^{-\frac{\varepsilon^2 W'}{2m}} + \frac{n+1}{T}) + O(\frac{err}{T})$, and this is $o(1)$ under the assumption of Theorem 2.

5 Lower Bound

In this section, we prove a lower bound on R_T^μ and R_T.

Theorem 4. *For bandit max-min fair allocation problem with $m \geq n$, the surrogate regret R_T^μ of any algorithm is at least $\Omega(m\sqrt{T}/n)$.*

If $T \geq \max\{n, m^2\} \geq 2$ and $m/n \geq \lceil 2338 \ln T \rceil$ in addition, then the regret R_T is also at least $\Omega(m\sqrt{T}/n)$.

We first prove this for any deterministic algorithm based on the idea of [5,4], and then extend the proof to any randomized algorithm. In the following, we use ALG to denote both an algorithm and its reward.

Fix any deterministic algorithm. Let b be a positive integer and $m = nb$. We use two index (j,k) ($j = 1,\dots,n$ and $k = 1,\dots,b$) to represent one item e. An item (j,k) is called the j-th item in the k-th item block k. Then, we can write the set of allocations as follows:

$$\mathcal{A} = \left\{a \in \{0,1\}^{n\times n\times b} : \textstyle\sum_i a_{i,j,k} = 1 \text{ for } \forall j,k\right\}.$$

Similarly, we define the set of optimal allocations as follows:

$$\mathcal{A}^* = \left\{a \in \{0,1\}^{n\times n\times b} \;:\; \begin{array}{l}\sum_i a_{i,j,k} = 1 \text{ for } \forall j,k \\ \sum_j a_{i,j,k} = 1 \text{ for } \forall i,k\end{array}\right\}.$$

For any $\alpha \in \mathcal{A}^*$, $j \in [n]$ and $k \in [b]$, let $I_{\alpha,j,k}$ be the unique i such that $\alpha_{i,j,k} = 1$.

Now we design a hard instance for the problem. Let $\varepsilon \in (0,1)$ be a parameter. We first choose $\alpha \in \mathcal{A}^*$ arbitrarily and set a distribution $D_{i,j,k}$ (of agent i for item (j,k)) to be a Bernoulli distribution $\mathrm{Ber}\,(1/2 + \varepsilon\alpha_{i,j,k})$. We refer to $(\mathrm{Ber}\,(1/2+\varepsilon\alpha_{i,j,k}))_{i,j,k}$ as α-adversary. Moreover, for each $\alpha \in \mathcal{A}^*$ and $k' \in [b]$, we also define another adversary called $(\alpha - k')$-adversary as follows: $D_{i,j,k} = \mathrm{Ber}\,(1/2)$ if $k = k'$, and $D_{i,j,k} = \mathrm{Ber}\,(1/2+\varepsilon\alpha_{i,j,k})$ otherwise. Note that for an allocation $\beta \in \mathcal{A}^*$, the $(\alpha - k')$-adversary is the same as the $(\beta - k')$-adversary if $\alpha_{i,j,k} = \beta_{i,j,k}$ for each $i \in [n]$, $j \in [n]$ and $k \in [b] \setminus \{k'\}$. When we use an α-adversary, for each agent i and each item (j,k), we say that (i,j,k) is a *correct assignment* if $\alpha_{i,j,k} = 1$. We use $\mathbb{P}_\alpha[\cdot]$ and $\mathbb{E}_\alpha[\cdot]$ to denote the conditional probability and expectation when we choose an α-adversary at first.

Let $\alpha^* \in \mathcal{A}^*$ be the most unfavorable adversary that minimize the reward. Let $\mu_{i,j,k}$ be the expected value of each $D_{i,j,k}$. We denote $N_{\alpha,k} \coloneqq \sum_{t,i,j} \alpha_{i,j,k} a^t_{i,j,k}$. Then we have

$$\begin{aligned}\mathbb{E}_{\alpha^*}[\mathrm{ALG}_\mu] &= \mathbb{E}_{\alpha^*}\left[\min_i \sum_{t,j,k} \mu_{i,j,k} a^t_{i,j,k}\right] \le \frac{1}{n|\mathcal{A}^*|}\sum_{\alpha\in\mathcal{A}^*}\mathbb{E}_\alpha\left[\sum_{t,i,j,k}\mu_{i,j,k}a^t_{i,j,k}\right] \\ &= \frac{1}{2}bT + \frac{\varepsilon}{n|\mathcal{A}^*|}\sum_{k=1}^{b}\sum_{\alpha\in\mathcal{A}^*}\mathbb{E}_\alpha[N_{\alpha,k}], \qquad (7)\end{aligned}$$

where we substitute $\mu_{i,j,k} = \frac{1}{2} + \varepsilon\alpha_{i,j,k}$ in the last equality. Next, we show the following lemma.

Lemma 5. *For each* $0 < \varepsilon \le 1/4$, $\mathbb{E}_\alpha[N_{\alpha,k}] \le \mathbb{E}_{\alpha-k}[N_{\alpha,k}] + 2\varepsilon nT\sqrt{\mathbb{E}_{\alpha-k}[N_{\alpha,k}]}$.

Proof. Let $\sigma^t \in \{0,1\}^{n\times b}$ denote the feedback that the algorithm observes at round t, i.e., the (j,k) entry of σ^t is $v^t_{i',j,k}$ where i' is the agent who receives (j,k). In addition, for $t = 1,\dots,T$, we denote by $S_t = (\sigma^1,\dots,\sigma^t) \in \{0,1\}^{n\times b\times t}$ all feedback observed up to round t. In the rest of the proof, we use the following notation on the KL-divergence:

$$K_t \coloneqq \sum_{S_t\in\{0,1\}^{n\times b\times t}} \mathbb{P}_{\alpha-k}[S_t]\ln\frac{\mathbb{P}_{\alpha-k}[S_t]}{\mathbb{P}_\alpha[S_t]},$$

$$K'_t \coloneqq \sum_{S_t \in \{0,1\}^{n \times b \times t}} \mathbb{P}_{\alpha-k}\left[S_t\right] \ln \frac{\mathbb{P}_{\alpha-k}\left[\sigma^t | S_{t-1}\right]}{\mathbb{P}_{\alpha}\left[\sigma^t | S_{t-1}\right]}.$$

By the chain rule, we have $K_T = \sum_{t=1}^{T} K'_t$. Since the algorithm is assumed to be deterministic, we can treat $N_{\alpha,k}$ as a function f of S_T. Then, the following holds:

$$\begin{aligned}
\mathbb{E}_{\alpha}\left[N_{\alpha,k}\right] - \mathbb{E}_{\alpha-k}\left[N_{\alpha,k}\right] &= \mathbb{E}_{\alpha}\left[f(S_T)\right] - \mathbb{E}_{\alpha-k}\left[f(S_T)\right] \\
&= \sum_{S_T} f(S_T)(\mathbb{P}_{\alpha}\left[S_T\right] - \mathbb{P}_{\alpha-k}\left[S_T\right]) \\
&\leq \sum_{S_T : \mathbb{P}_{\alpha}[S_T] > \mathbb{P}_{\alpha-k}[S_T]} f(S_T)(\mathbb{P}_{\alpha}\left[S_T\right] - \mathbb{P}_{\alpha-k}\left[S_T\right]) \\
&\leq nT \sum_{S_T : \mathbb{P}_{\alpha}[S_T] > \mathbb{P}_{\alpha-k}[S_T]} (\mathbb{P}_{\alpha}\left[S_T\right] - \mathbb{P}_{\alpha-k}\left[S_T\right]) \\
&= \frac{nT}{2} \sum_{S_T} |\mathbb{P}_{\alpha}\left[S_T\right] - \mathbb{P}_{\alpha-k}\left[S_T\right]| \\
&\leq \frac{nT}{2} \sqrt{2K_T} = \frac{nT}{2} \sqrt{2 \textstyle\sum_{t=1}^{T} K'_t}, \qquad (8)
\end{aligned}$$

where $N_{\alpha,k} \coloneqq f(S_T)$, the first inequality is due to $N_{\alpha,k} \leq nT$ and the last inequality is due to the Pinsker's inequality. K'_t is computed as follows.

Claim. Fix any S_{t-1} and let $P(S_{t-1})$ be the number of correct assignments (i, j, k') in a^t such that $k' = k$, i.e., $P(S_{t-1}) \coloneqq \sum_{i,j=1}^{n} \alpha_{i,j,k} a^t_{i,j,k}$. Then, we have $K'_t = \frac{1}{2} \ln \frac{1}{1-4\varepsilon^2} \mathbb{E}_{\alpha-k}\left[P(S_{t-1})\right]$.

The proof of the claim can be found in Claim 3 of the full version [27]. By applying the claim to (8), it follows that

$$\begin{aligned}
\mathbb{E}_{\alpha}\left[N_{\alpha,k}\right] - \mathbb{E}_{\alpha-k}\left[N_{\alpha,k}\right] &\leq \frac{nT}{2} \sqrt{\ln \frac{1}{1-4\varepsilon^2} \sum_{t=1}^{T} \mathbb{E}_{\alpha-k}\left[P(S_{t-1})\right]} \\
&= \frac{nT}{2} \sqrt{\ln \frac{1}{1-4\varepsilon^2} \mathbb{E}_{\alpha-k}\left[N_{\alpha,k}\right]} \leq 2\varepsilon nT \sqrt{\mathbb{E}_{\alpha-k}\left[N_{\alpha,k}\right]},
\end{aligned}$$

where the last inequality follows from the convexity of $-\ln(1-x)$ and $0 < \varepsilon \leq 1/4$. This completes the proof.

By the definition of an $(\alpha - k)$-adversary, we obtain

$$\begin{aligned}
\sum_{\alpha \in \mathcal{A}^*} \mathbb{E}_{\alpha-k}\left[N_{\alpha,k}\right] &= \frac{1}{n!} \sum_{\beta \in \mathcal{A}^*} \sum_{\alpha : (\alpha-k) = (\beta-k)} \mathbb{E}_{\alpha-k}\left[N_{\alpha,k}\right] \\
&= \frac{1}{n!} \sum_{\beta \in \mathcal{A}^*} \mathbb{E}_{\beta-k} \left[\sum_{\alpha : (\alpha-k) = (\beta-k)} \sum_{t,i,j} a^t_{i,j,k} \alpha_{i,j,k} \right]
\end{aligned}$$

$$= \frac{1}{n!} \sum_{\beta \in \mathcal{A}^*} \mathbb{E}_{\beta-k} \left[\sum_{t,i,j} a^t_{i,j,k} \sum_{\alpha:(\alpha-k)=(\beta-k)} \alpha_{i,j,k} \right]$$
$$= \frac{1}{n!} \sum_{\beta \in \mathcal{A}^*} (nT \cdot (n-1)!) = |\mathcal{A}^*|T.$$

By this result, Lemma 5 and the Cauchy-Schwartz inequality, it follows that

$$\textstyle\sum_\alpha \mathbb{E}_\alpha [N_{\alpha,k}] \leq |\mathcal{A}^*|T + 2\varepsilon nT\sqrt{|\mathcal{A}^*| \cdot |\mathcal{A}^*|T}$$

and then, $\frac{\varepsilon}{n|\mathcal{A}^*|} \sum_{k=1}^{b} \sum_{\alpha \in \mathcal{A}^*} \mathbb{E}_\alpha [N_{\alpha,k}] \leq \varepsilon bT \left(\frac{1}{n} + 2\varepsilon\sqrt{T} \right)$. Note that $\mathrm{OPT}_\mu = (1/2 + \varepsilon)bT$. By tuning $\varepsilon = 1/(8\sqrt{T})$, we finally have the following lower bound:

$$\mathrm{OPT}_\mu - \mathbb{E}_{\alpha^*} [\mathrm{ALG}_\mu] \geq \varepsilon bT \left(1 - \frac{1}{n} - 2\varepsilon\sqrt{T} \right) \geq \frac{1}{32} b\sqrt{T} = \Theta\left(\frac{m}{n}\sqrt{T} \right). \quad (9)$$

By Yao's principle, the lower bound also applies to randomized algorithms. This concludes the proof of the first part of Theorem 4.

Lower Bound for R_T

Next we show a lower bound on R_T. Under the assumptions $T \geq \max\{n, m^2\}$ and $m/n \geq \lceil 2338 \ln T \rceil$, we claim that OPT_μ is close enough to $\mathbb{E}[\mathrm{OPT}]$, whose proof is found in Lemma 6 of the full version [27].

Claim. $\mathrm{OPT}_\mu \leq \mathbb{E}[\mathrm{OPT}] + (\frac{1}{32} - \frac{1}{1000})b\sqrt{T}$.

Since we can show that $\mathbb{E}_{\alpha^*}[\mathrm{ALG}] \leq \frac{1}{2}bT + \frac{\varepsilon}{n|\mathcal{A}^*|} \sum_{k=1}^{b} \sum_{\alpha \in \mathcal{A}^*} \mathbb{E}_\alpha [N_{\alpha,k}]$ in a way similar to (7), the lower bound $b\sqrt{T}/32$ established in the first part of Theorem 4 is also a lower bound of $\mathrm{OPT}_\mu - \mathbb{E}[\mathrm{ALG}]$. This holds also for randomized algorithms. Plugging the clam into (9), we finally see that $\mathbb{E}[\mathrm{OPT}] - \mathbb{E}_{\alpha^*}[\mathrm{ALG}] \geq \frac{1}{1000} b\sqrt{T} = \Theta\left(\frac{m}{n}\sqrt{T} \right)$. Then we see that the second part of Theorem 4 holds.

6 Conclusion and Discussion

In this paper, we introduced the bandit max-min fair allocation problem. We have proposed an algorithm with a regret bound of $O(m\sqrt{T}\ln T/n + m\sqrt{T\ln(mnT)})$ when T is sufficiently large, and showed a lower bound $\Omega(m\sqrt{T}/n)$ on the regret. Thus, when T is sufficiently large, the bounds matches up to a logarithmic factor of T.

We remark that the regret bounds also apply to variations of our problem. One such case is maximizing the minimum "expected" utility: $\mathrm{ALG}_\mathrm{E} = \min_i \mathbb{E}\left[\sum_{t,e} \mu_{ie} a^t_{ie}\right]$ where the regret is defined as $\mathrm{OPT}_\mu - \mathrm{ALG}_\mathrm{E}$. In this setting, we have $\mathrm{ALG}_\mathrm{E} \geq \mathrm{ALG}_\mu$ and then similar proofs work to derive the same bounds.

For another, our algorithm works even when each agent's bundle in each round must satisfy a matroid constraint. We detail this in the full version [27].

One future work is to close the gap between the upper and lower bounds on R_T. An upper bound with a weaker assumption on T and a lower bound using $\mathbb{E}[\mathrm{OPT}]$ directly are also open. Another potential future work is to extend the problem setting to reflect practical situations. For example, in a subscription service, the rental period can be different depending on situations. It would be possible to improve a regret if users let us know what they probably dislike (i.e., item e with μ_{ie} being almost zero). We believe that such an extension of the problem provides insight into real-world applications.

Acknowledgments. This work was partially supported by the joint project of Kyoto University and Toyota Motor Corporation, titled "Advanced Mathematical Science for Mobility Society", JST ERATO Grant Number JPMJER2301, JST ASPIRE Grant Number JPMJAP2302, and JSPS KAKENHI Grant Numbers JP21K17708, JP21H03397, and JP25K00137.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Aleksandrov, M., Walsh, T.: Online fair division: A survey. Proceedings of the AAAI Conference on Artificial Intelligence **34**(09), 13557–13562 (2020)
2. Arora, S., Hazan, E., Kale, S.: The multiplicative weights update method: A meta-algorithm and applications. Theory of Computing **8**(1), 121–164 (2012)
3. Asadpour, A., Saberi, A.: An approximation algorithm for max-min fair allocation of indivisible goods. SIAM Journal on Computing **39**(7), 2970–2989 (2010)
4. Audibert, J.Y., Bubeck, S., Lugosi, G.: Regret in online combinatorial optimization. Mathematics of Operations Research **39**(1), 31–45 (2014)
5. Auer, P., Cesa-Bianchi, N., Freund, Y., Schapire, R.E.: The nonstochastic multiarmed bandit problem. SIAM Journal on Computing **32**(1), 48–77 (2002)
6. Badanidiyuru, A., Kleinberg, R., Slivkins, A.: Bandits with knapsacks. Journal of the ACM **65**(3), 13:1–13:55 (2018)
7. Banerjee, S., Gkatzelis, V., Gorokh, A., Jin, B.: Online Nash social welfare maximization with predictions. In: Proceedings of the 2022 Annual ACM-SIAM Symposium on Discrete Algorithms. pp. 1–19 (2022)
8. Bansal, N., Sviridenko, M.: The Santa Claus problem. In: Proceedings of the 38th Annual ACM Symposium on Theory of Computing. pp. 31–40 (2006)
9. Barman, S., Khan, A., Maiti, A.: Universal and tight online algorithms for generalized-mean welfare. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 36, pp. 4793–4800 (2022)
10. Benade, G., Kazachkov, A.M., Procaccia, A.D., Psomas, C.A.: How to make envy vanish over time. In: Proceedings of the 2018 ACM Conference on Economics and Computation. pp. 593–610 (2018)
11. Besson, L., Kaufmann, E.: What doubling tricks can and can't do for multi-armed bandits. arXiv preprint arXiv:1803.06971 (2018)
12. Bezáková, I., Dani, V.: Allocating indivisible goods. ACM SIGecom Exchanges **5**(3), 11–18 (2005)
13. Bistritz, I., Baharav, T.Z., Leshem, A., Bambos, N.: One for all and all for one: Distributed learning of fair allocations with multi-player bandits. IEEE Journal on Selected Areas in Information Theory **2**(2), 584–598 (2021)

14. Boursier, E., Perchet, V.: A survey on multi-player bandits. Journal of Machine Learning Research **25**(137), 1–45 (2024)
15. Bouveret, S., Chevaleyre, Y., Maudet, N.: Fair allocation of indivisible goods. In: Brandt, F., Conitzer, V., Endriss, U., Lang, J., Procaccia, A.D. (eds.) Handbook of Computational Social Choice, chap. 12, pp. 284–310. Cambridge University Press (2016)
16. Cesa-Bianchi, N., Lugosi, G.: Prediction and playing games. In: Prediction, Learning, and Games, chap. 7, pp. 180–232. Cambridge University Press (2006)
17. Chakrabarty, D., Chuzhoy, J., Khanna, S.: On allocating goods to maximize fairness. In: Proceedings of the 50th Annual IEEE Symposium on Foundations of Computer Science. pp. 107–116 (2009)
18. Chen, W., Wang, Y., Yuan, Y.: Combinatorial multi-armed bandit: General framework and applications. In: Proceedings of the 30th International Conference on Machine Learning. pp. 151–159 (2013)
19. Chen, Y., Cuellar, A., Luo, H., Modi, J., Nemlekar, H., Nikolaidis, S.: The fair contextual multi-armed bandit. In: Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems. pp. 1810–1812 (2020)
20. Claure, H., Chen, Y., Modi, J., Jung, M., Nikolaidis, S.: Multi-armed bandits with fairness constraints for distributing resources to human teammates. In: Proceedings of the 2020 ACM/IEEE International Conference on Human-Robot Interaction. pp. 299–308 (2020)
21. Cohen, S., Agmon, N.: Near-optimal online resource allocation in the random-order model. In: Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems. pp. 2219–2221 (2024)
22. Devanur, N.R., Jain, K., Sivan, B., Wilkens, C.A.: Near optimal online algorithms and fast approximation algorithms for resource allocation problems. Journal of the ACM **66**(1), 7:1–7:41 (2019)
23. Feige, U.: On allocations that maximize fairness. In: Proceedings of the 19th Annual ACM-SIAM Symposium on Discrete Algorithms. pp. 287–293 (2008)
24. Golovin, D.: Max-min fair allocation of indivisible goods. Technical Report CMU-CS-05-144, Carnegie Mellon University (June 2005)
25. Haeupler, B., Saha, B., Srinivasan, A.: New constructive aspects of the lovász local lemma. Journal of the ACM **58**(6), 28:1–28:28 (2011)
26. Hajiaghayi, M., Khani, M., Panigrahi, D., Springer, M.: Online algorithms for the Santa Claus problem. In: Advances in Neural Information Processing Systems 35. vol. 35, pp. 30732–30743 (2022)
27. Harada, T., Ito, S., Sumita, H.: Bandit max-min fair allocation. arXiv preprint arXiv:2505.05169 (2025)
28. Hossain, S., Micha, E., Shah, N.: Fair algorithms for multi-agent multi-armed bandits. Advances in Neural Information Processing Systems **34**, 24005–24017 (2021)
29. Igarashi, A., Lackner, M., Nardi, O., Novaro, A.: Repeated fair allocation of indivisible items. In: Proceedings of the 38th AAAI Conference on Artificial Intelligence. pp. 9781–9789 (2024)
30. Jones, M., Nguyen, H., Nguyen, T.: An efficient algorithm for fair multi-agent multi-armed bandit with low regret. Proceedings of the AAAI Conference on Artificial Intelligence **37**(7), 8159–8167 (2023)
31. Kawase, Y., Sumita, H.: Online max-min fair allocation. In: Proceedings of International Symposium on Algorithmic Game Theory. pp. 526–543 (2022)
32. Lai, T., Robbins, H.: Asymptotically efficient adaptive allocation rules. Advances in Applied Mathematics **6**(1), 4–22 (1985)
33. Lenstra, J.K., Shmoys, D.B., Tardos, É.: Approximation algorithms for scheduling unrelated parallel machines. Mathematical Programming **46**, 259–271 (1990)

34. Li, F., Liu, J., Ji, B.: Combinatorial sleeping bandits with fairness constraints. IEEE Transactions on Network Science and Engineering **7**(3), 1799–1813 (2019)
35. Micheel, K.J., Wilczynski, A.: Fairness in repeated house allocation. In: Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems. pp. 2390–2392 (2024)
36. Patil, V., Ghalme, G., Nair, V., Narahari, Y.: Achieving fairness in the stochastic multi-armed bandit problem. Journal of Machine Learning Research **22**(174), 1–31 (2021)
37. Zhang, M., Deo-Campo Vuong, R., Luo, H.: No-regret learning for fair multi-agent social welfare optimization. In: Advances in Neural Information Processing Systems. vol. 37, pp. 57671–57700 (2024)